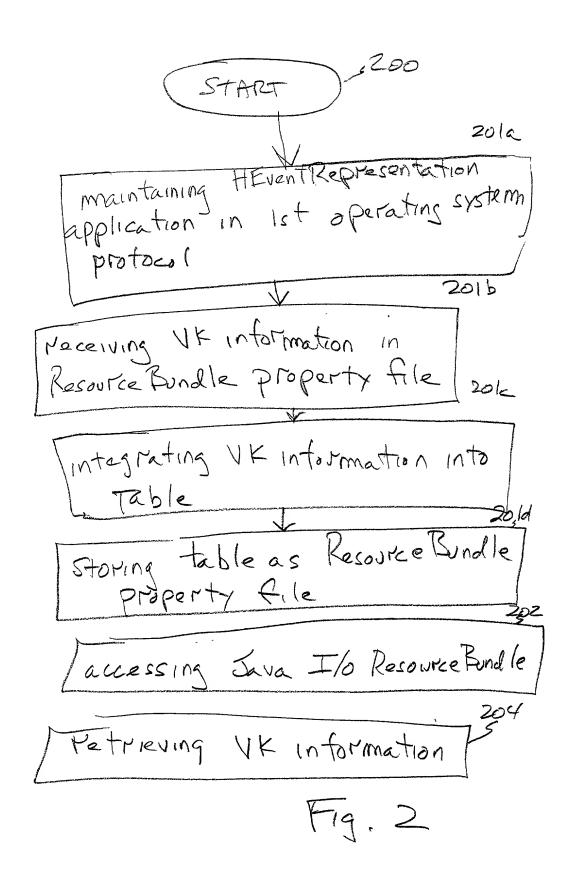
(CHA C+) = 100
START) 5100
10,121
/receiving VK information as
Java source code 101a2
receiving HEvent Representation application
as Sava Source Code
V
Empling Java source codes into JMV 510161
legte codes
10/21
Carchiving JMV Gytecodes into JAR file?
TEAT ENTO GIVE GOVES THE
10102
laccessing a JAR file /s 102
104
Tetreiving virtual key information
r _{9.} (
\mathcal{J} .



START J-300
SO 1a
l'eceiving storage driver as machine codes
30/6
Storing Storage driver as machine codes
receiving Uk information as binary code
Using storage d'river to cross-Peterence
Using storage driver to cross-reference VK information with ERROM addresses 301e
Totoring VK information as machine codes)
202
Calling JNI 75302
Using Java byte codes to call storage driver 306
driver 306
Jaccessing mapped memory
308
Tretrieving VK information
F.q. 3

```
[**platform event representation - color]
VK_GO_TO_START_Color = 21, 21,
VK_REWIND_Color = 0, 0, 0
VK_STOP_Color = 0, 0, 0
VK_PAUSE_Color = 0, 0, 0
VK_PAUSE_Color = 0, 0, 0
VK_PAT_FAST_FWD_Color = 0, 0, 0
VK_GO_TO_END_Color = 0, 0, 0
VK_TRACK_PREV_Color = 0, 0, 0
VK_TRACK_NEXT_Color = 0, 0, 0
VK_RECORD_Color = 245, 0, 0
VK_EST_TOGGLE_Color = 0, 0, 0
VK_USE_USE_TOGGLE_Color = 0, 0, 0
VK_VOLUME_UP_Color = 0, 0, 0
VK_VOLUME_DOWN_Color = 0, 0 0
VK_UP_Color = 0, 0, 0
           VK_UP_Color = 0, 0, 0
VK_DOWN_Color = 0, 0, 0
VK_LEFT_Color = 0, 0, 0
VK_RIGHT_Color = 0, 0, 0
VK_POWER_Color = 0, 0, 0
[**platform event representation - String]

VK_GO_TO_START_String = Two equilateral triangles, pointing at a line to the left VK_REWIND_String = Two equilateral triangles, pointing to the left VK_STOP_String = Two equilateral triangles, pointing to the right VK_PAUSE_String = Two equilateral triangles, pointing to the right VK_PAST_FWD_String = Two equilateral triangles, pointing to a line at the right VK_GO_TO_END_String = Two equilateral triangles, pointing to a line at the left VK_TRACK_PREV_String = One equilateral triangle, pointing to a line at the left VK_TRACK_NEXT_String = One equilateral triangle, pointing to a line at the right VK_RECORD_String = One equilateral triangle, pointing to a line at the right VK_RECORD_String = A carde, normally red
VK_ETCT_TOGGLE_String = A line under a wide triangle which points up VK_OLUME_DUP_String = A ramp, increasing to the right, near a minus sign VK_UP_String = An arrow pointing up
VK_DUME_DUMN_String = An arrow pointing down
VK_ETCT_String = An arrow pointing to the right

= An arrow pointing to the right

= An arrow pointing to the right

= A circle, broken at the top, with a vertical line in the break
   [**platform event representation - image]
VK GO TO START Image = start png
VK REWIND Image = rewind png
VK STOP Image = stopping
VK PAUSE Image = play png
VK PALY Image = play png
VK FAST FWD Image = restifted png
VK GO TO END Image = restirted png
VK TRACK NEXT Image = nexitrack png
VVK TRACK NEXT Image = nexitrack png
VVK TRACK NEXT Image = record png
VK TRECORD Image = record png
VK EJECT TOGGLE Image = yelet png
VK VE UP Image = volup png
VK VOLUME DOWN Image = volup png
VK UP Image = up png
VK UP Image = up png
VK LEFT Image = up png
VK LEFT Image = left png
VK RIGHT Image = right png
VK RIGHT Image = right png
VK POWER Image = power png
[**platform event representation - type]

VK_GO_TO_START_TYPE = ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_REWIND_Type = ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_STOP_TYPE = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_PAUSE_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_PLAY_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_PAST_FWD_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_GO_TO_END_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR |

ER_TYPE_SYMBOL

VK_TRACK_PREV_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR |

ER_TYPE_SYMBOL

VK_ECORD_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR |

ER_TYPE_SYMBOL

VK_ECORD_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR |

ER_TYPE_SYMBOL

VK_VOLUME_UP_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR |

ER_TYPE_SYMBOL

VK_VOLUME_UP_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR |

ER_TYPE_SYMBOL

VK_VOLUME_DOWN_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR |

ER_TYPE_SYMBOL

VK_DOWN_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_DOWN_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_DOWN_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_RIGHT_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_RIGHT_Type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_RIGHT_TYPE = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

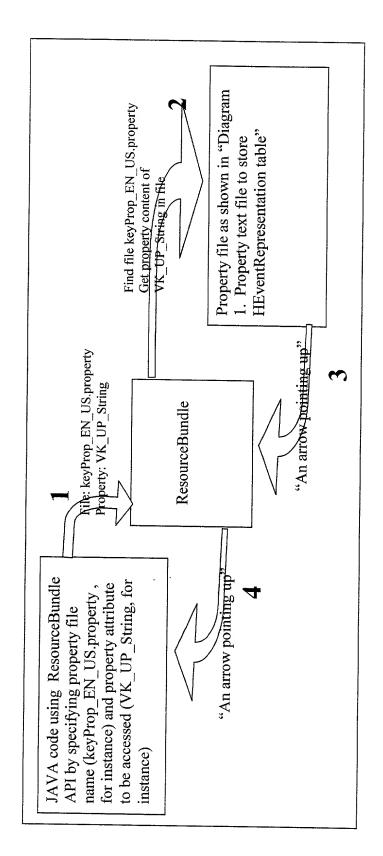
VK_RIGHT_TYPE = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_POWER_TIPE = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_POWER_TIPE = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL

VK_POWER_TIPE = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_
```

Fig. 4a Property text file to store HEventRepresentation table.



F19, 46

```
String[] eventRepresentationData = {
   VK_GO_TO_START, new Color(r, g, b), "|\", "start.png", ER_TYPE_STRING | ER_TYPE COLOR | ER TYPE SYMBOL.
   VK_GO_TO_REWIND, new Color(r, g, b), "", "rewind png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
   VK_GO_TO_STOP, new Color(r, g, b), "STOP", "stop png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
   VK_GO_TO_PAUSE, new Color(r, g, b), "| ", "pause.png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL.
   VK_GO_TO_PLAY, new Color(r, g, b), "play.png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL, VK_GO_TO_FAST_FWD, new Color(r, g, b), ">**, "fastfwd.png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
   VK_GO_TO_GO_TO_END new Color(r, g, b), ">", "end.png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE SYMBOL,
   VK_GO_TO_TRACK_PREV, new Color(r, g, b), "| <", "prevtrack.png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
   VK_GO_TO_TRACK_NEXT, new Color(r, g, b), ">", "nexttrack1.png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL.
   VK_GO_TO_RECORD, new Color(r, g, b), "O", "record.png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
   VK_GO_TO_EJECT_TOGGLE, new Color(r, g, b), "ELECT", "eject.png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
   VK\_GO\_TO\_VOLUME\_UP, new Color(r, g, b), \textbf{`VOL+''}, \quad \text{``volup png''}, ER\_TYPE\_STRING | ER\_TYPE\_COLOR | ER\_TYPE\_SYMBOL, \\
   VK_GO_TO_VOLUME_DOWN, new Color(r, g, b), "VOL-", "voldown.png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
   VK_GO_TO_UP, new Color(r, g, b), "A", "up png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
   VK_GO_TO_DOWN, new Color(r, g, b), "V", "down.png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
   VK_GO_TO_LEFT, new Color(r, g, b), "<", "left.png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
   VK_GO_TO_RIGHT, new Color(r, g, b), "->", "right png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
   VK_GO_TO_POWER, new Color(r, g, b), "0/1", "nght.png", ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
```

Fig. 5 Text array of key event representation

```
package com sharplabs havi unutil
      mport sava awt Color
  static class VK_REWIND {
static final int code = VK_REWIND
static final Color c = new Color(0, 0 0)
static final String s = "<<"
            static final String s="cv:nd-color, 0, 0 or static final String series and string series static final string series static final string imgFile = "rewind.png" static final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL
         static class VK_STOP
           }
state class VK_PAUSE;
state final int code = VK_PAUSE,
state final color complete = VK_PAUSE,
state final state final state final String s="limes right;
state final String imgFile = 'pause png'
state final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,

"The state final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
"The state final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
"The state final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
"The state final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
"The state final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
"The state final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
"The state final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
"The state final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
"The state final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
"The state final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
"The state final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_STR
        }
state class VK_PLAY {
state final int code = VK_PLAY,
state final clotr c = new Color(0, 0, 0)
state final String s = "">
state final String mgFile = "play png"
state final String type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
        } statuc class VK_FAST_FWD {
    statuc final mt code = VK_FAST_FWD
    statuc final color c = new Color(0, 0, 0)
    statuc final String s = ">"
    statuc final String magRue = "feat, two png",
    statuc final int type = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
         statuc class VK_TRACK_PREV {
    statuc final int code = VK_TRACK_PREV {
    statuc final statuc final color = o = new Color(0, 0, 0).
    statuc final String s= "|<".
    statuc final int statuc final int |
    ype = ER_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL. |
        }
state class VK_EJECT_TOGGLE {
state final int code = VK_EJECT_TOGGLE,
state final Color = c=new Color(0,0,0),
state final String = s="EJECT",
state final String immgFile = "eject ping",
state fina
           } static class VK_VOLUME_UP {
    static final int code = VK_VOLUME_UP,
    static final Color = c = new Colort(0, 0, 0)
    static final Color = c = new Colort(0, 0, 0)
    static final String = s = "VOL"*
    static final String imgFile = "Volumeup pag".
    static final String imgFile = "Volumeup pag".
    static final int = type = Er_TYPE_NOT_SUPPORTED | ER_TYPE_STRING | ER_TYPE_COLOR | ER_TYPE_SYMBOL,
```

Fig. 6 Static class of event representation

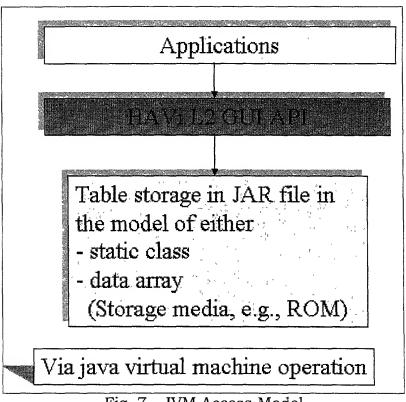


Fig. 7 JVM Access Model

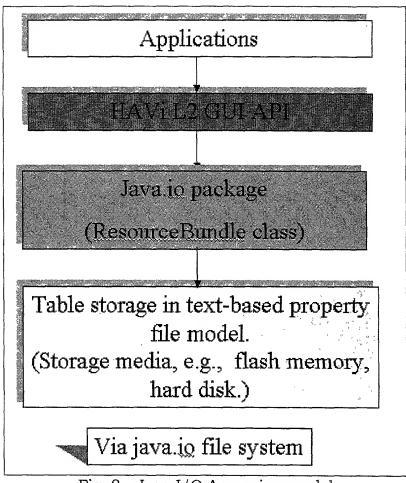


Fig. 8 Java I/O Accessing model

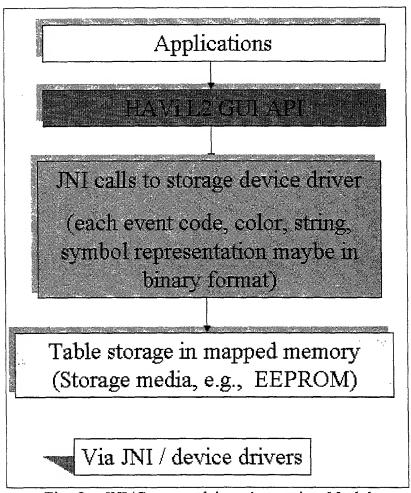


Fig. 9 JNI/Storage driver Accessing Model

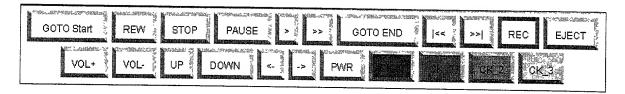


Fig. 10 HEventRepresentation using String, Color attribute data

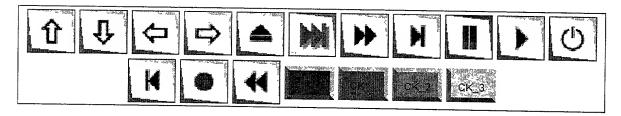


Fig. 11 HEventRepresentation using Symbol, String, and Color attribute data